(2)

AD-A240 048

N PAGE

Form Approved
OMB No. 0704-0188

1. AGER

3. REPORT TYPE AND DATES COVERED
FINAL, 01 APR 89 - 31 JUL 90

4. TITLE AND SUBTITLE
A GRID-FREE METHOD FOR HIGH REYNOLDS NUMBER FLOW AROUND AN IMMERSED ELASTIC STRUCTURE

5. FUNDING NUMBERS
AFOSR-89-0295

62202F, 6177 55

6. AUTHOR(S)
Lisa J. Fauci

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Tulane University--Uptown Campus
6823 St. Charles Ave.
New Orleans, LA 70118

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR 91 0730

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
Building 410
Bolling Air Force Base
Washington, DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFOSR-89-0295

DTIC
ELECTE
SEP 06 1991
S D
D

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release;
distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Finite difference schemes do not perform well in simulations of high Reynolds number flow because a restrictive number of grid points must be used to resolve boundary layers. In this report we present a grid-free numerical method which may be used to calculate flows around an immersed elastic structure. Numerical results are presented in two simple cases: flow past a circular cylinder and flow past a flat plate. Here the boundary motion is specified and is not time dependent. However, this algorithm can be used to simulate more complicated, time dependent problems in biological fluid mechanics.

14. SUBJECT TERMS

91 9 5 008

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

A Grid-free Method for High Reynolds Number Flow Around an Immersed Elastic Structure

Lisa J. Fauci

Department of Mathematics

Tulane University

New Orleans, Louisianna

Final Technical Report: AFOSR-89-0295

Abstract

Finite difference schemes do not perform well in simulations of high Reynolds number flow because a restrictive number of grid points must be used to resolve boundary layers. In this report we present a grid-free numerical method which may be used to calculate flows around an immersed elastic structure. Numerical results are presented in two simple cases: flow past a circular cylinder and flow past a flat plate. Here the boundary motion is specified and is not time dependent. However, this algorithm can be used to simulate more complicated, time dependent problems in biological fluid mechanics.

1. Introduction

This project concerns the computational modeling of problems in biological fluid mechanics. Typically, these involve the study of flow about highly flexible, moving boundaries; for instance, distensible vessels, moving heart walls and valves. The fluid along with the boundary constitues a coupled mechanical system in the sense that the motion of the boundary is determined by that of the fluid, but at the same time the boundary exerts force on the fluid and alters its motion. With the advent of supercomputers, many of these problems which were previously intractable, can now be successfully analyzed.

An innovative computational approach, the *immersed boundary technique*, was introduced by Peskin [16] to model two-dimensional blood flow in the left heart. Recently, this technique has been advanced to study other biofluiddynamic problems such as platelet aggregation [10] aquatic animal locomotion [7][9], peristaltic pumping of solid particles [8], fluid flow in the inner ear [3], and three-dimensional blood flow in the heart [17][18]. This numerical method solves the full incompressible Navier-Stokes equations in a domain of fluid within which a massless, neutrally-buoyant elastic boundary undergoing time dependent movements is immersed.

The Navier-Stokes equations have been solved using Chorin's finite difference scheme [5]. Theoretically, this scheme imposes no restriction on the Reynolds number of the flow to be modelled. (The Reynolds number is a nondimensional quantity which measures the ratio of inertial forces to viscous forces.) However, the higher the Reynolds number, the smaller the grid spacing and time step must be to ensure stability of the calculations. To model high Reynolds number flow (as in the heart), the amounts of computer time and memory needed become prohibitive and impractical.

Within the finite difference framework, we have adapted the second order projection method recently introduced by Bell, Colella, and Glaz [2] for the solution to the Navier-Stokes equations. Our results indicate that flows with higher Reynolds numbers can be modelled with this code than with Chorin's scheme using the same number of grid points

and the same time step. We have tested this scheme on flow past a tethered cylinder and have exhibited flow separation and vortex shedding. The cylinder is modelled as an immersed boundary. Samn [20] has tested the projection method on problems with a non-zero force density, as is the case of immersed boundary calculations, in a class of problems where exact solutions are known. For moderately high Reynolds numbers, this projection method is quite promising.

For much higher Reynolds numbers, a grid-free vortex method combined with the immersed boundary technique may be more appropriate. An effort in this direction was made by McCracken and Peskin [15] in 1980. This was a hybrid method which represented vorticity both on a grid and as free-moving vortex elements. In light of the advances made in vortex methods and the incredible growth of computing power since then, a totally grid-free vortex method for two-dimensional flow is now being implemented and tested. This report will describe the current status of this algorithm.

In section 2, we will briefly discuss the immersed boundary technique, and in section 3 we will describe vortex methods. In section 4 we will present our grid-free vortex method and discuss how it treats an immersed boundary. Examples of flow past a cylinder and vortex roll-up due to the motion of a flat plate will be presented in Section 5. Finally in Section 6 we will discuss the issues which need to be addressed to make this method robust and competitive.

2. Immersed Boundary Technique

The flow of a viscous, incompressible fluid is governed by the incompressible Navier Stokes equations:

$$\begin{aligned} \rho\left[\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right] &= -\nabla p + \mu\Delta\mathbf{u} + \mathbf{F}(\mathbf{x},t) \\ \nabla\cdot\mathbf{u} &= 0 \end{aligned} \tag{1}$$

Here ρ = density, μ = viscosity, $\mathbf{u} = (u,v)$ = velocity, p = pressure, and $\mathbf{F}(\mathbf{x},t)$ is the external force per unit volume applied to the fluid. This external force field represents the force of the elastic boundary on the fluid. It is a delta-function layer of force which is zero away from the immersed boundary $\mathbf{X}(s,t)$. This representation will be the basis of the computational model.

Since the immersed boundary is taken to be elastic and massless, the density of the boundary force $\mathbf{f}(s,t)$ at a material point is determined by the boundary configuration at time t. We will discuss the derivation of the boundary force below. This elastic force is transmitted directly to the fluid because the boundary is massless. This follows from Newton's second law for force balance at a boundary point: the force of the fluid on the boundary point and the elastic force on the boundary point must add up to zero. Therefore, the force of the boundary point on the fluid is equal to the elastic force on that point. The external force field is then:

$$\mathbf{F}(\mathbf{x},t) = \int \mathbf{f}(s,t)\delta(\mathbf{x} - \mathbf{X}(s,t))ds \tag{2}$$

Here the integration is over the immersed boundary, and δ is the two-dimensional delta-function.

Since the fluid is viscous, the velocity field is continuous across the boundary. This implies that the velocity of a material point of the immersed boundary must be equal to the velocity of the fluid at that point:

$$\frac{\partial \mathbf{X}(s,t)}{\partial t} = \mathbf{u}(\mathbf{X}(s,t),t) = \int \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(s,t))d\mathbf{x} \tag{3}$$

Here the integration is over the entire fluid domain. This integral representation is not singular since the two-dimensional delta-function is integrated over both space dimensions.

The crucial feature of this model is that the immersed boundary is not our computational boundary in the Navier-Stokes solver (whether we are using finite differences or vortex methods). It is a singular force field which alters the driving force in the fluid dynamic equations.

Algorithm

Given the fluid velocity field $\mathbf{u}^n$ and the configuration of the elastic boundaries $\mathbf{X}^n$ at time step n:

1) Calculate the force density $\mathbf{f}^n$ from the configuration $\mathbf{X}^n$.

2) Use $\mathbf{f}^n$ to determine the external force on the fluid: $\mathbf{F}^n$.

3) Solve the Navier-Stokes equations for $\mathbf{u}^{n+1}$.

4) Move the elastic boundary at the local fluid velocity to get $\mathbf{X}^{n+1}$.

Within the finite difference framework, step (3) has been solved using Chorin's finite difference method. Steps (2) and (4) involve the use of discrete delta functions which communicate information between the grid and the immersed boundary points (which do not coincide with grid points). We refer the reader to [16] for details.

Derivation of force density

We will first model a *stationary* immersed boundary in order to illustrate our choice of force density. Consider a boundary made up of a continuum of material points $\mathbf{X}(s,t) = (x(s,t), y(s,t))$ which is in equilibrium when its material points coincide with the tether points $\mathbf{X}^*(s) = (x^*(s), y^*(s))$. We imagine there is a spring connecting $\mathbf{X}(s,t)$ to $\mathbf{X}^*(s)$ and that it has resting length zero.

The elastic force acting on a point $\mathbf{X}(s,t)$ is then:

$$\mathbf{f}(s,t)ds = -S[\mathbf{X}(s,t) - \mathbf{X}^*(s)]ds \tag{4}$$

S is the stiffness constant of the spring, and it may be thought of in two ways:

1) A physical parameter which reflects material properties of the elastic boundary.

2) A numerical parameter which should be made as large as possible to strictly enforce the equilibrium position of the elastic boundary.

This representation of the force density assumes that we know a priori the equilibrium position of the boundary in fixed space. This will be useful in our test examples where our immersed boundary will be either a fixed cylinder or a fixed flat plate. However, in most applications, the motion which we choose to specify is the time-dependent configuration of the boundary with respect to itself. The actual displacement relative to spatial coordinates is not preset, but is determined by the equations of motion. For instance, we can choose forces at a boundary point due to the rest of the boundary as:

1) Spring-like forces which asks that the "links" between successive points resist compression or expansion from a given arclength.

2) Bending-resistant forces which asks that the angle formed by neighboring links be a given function which changes with position and time.

We refer the reader to [9][16] for more details on how different boundary configurations can be enforced by the proper choice of force density functions.

3. Vortex Method

In this section, we will outline Chorin's vortex method [6], a grid-free scheme which represents the fluid field by a discrete collection of vortex elements.

Inviscid, incompressible flow in two dimensions in the absence of external forces is governed by the Euler equations:

$$\frac{D\mathbf{u}}{Dt} = -\nabla p \tag{5}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{6}$$

The first term in equation (5) is the total derivative. We introduce the vorticity $\omega = \nabla \times \mathbf{u} \cdot (0,0,1) = v_x - u_y$. Taking curl of equation (5) gives:

$$\frac{D\omega}{Dt} = 0 \tag{7}$$

The vorticity acts as "particles" which move with the fluid. Since this flow is inviscid, there is no mechanism for creation or destruction of vorticity (Kelvin's circulation theorem). The vorticity is represented by a sum of m point vortices, the i th one centered at point $\mathbf{x}_i$ with strength κ_i:

$$\omega = \sum_{i=1}^{m} \omega_i = \sum_{i=1}^{m} \kappa_i \delta(\mathbf{x} - \mathbf{x}_i) \tag{8}$$

Here δ is the Dirac delta function.

The incompressibility condition implies the existence of a stream function $\psi(x,y)$ such that:

$$\psi_y = u \tag{9}$$

$$\psi_x = -v \tag{10}$$

The relation between the stream function and the vorticity is therefore:

$$\Delta\psi = -\omega \tag{11}$$

Using the representation (8) of vorticity, equation (11) can be solved explicitly (in the whole plane) by the formula:

$$\psi(\mathbf{x}) = \sum_{i=1}^{m} \frac{\kappa_i}{2\pi} log|\mathbf{x} - \mathbf{x}_i| \tag{12}$$

The velocity can be calculated directly from the stream function:

$$\mathbf{u}_\omega(\mathbf{x}) = \sum_{i=1}^{m} \frac{\kappa_i}{2\pi} \frac{(-(y - y_i), (x - x_i))}{|\mathbf{x} - \mathbf{x}_i|^2} \tag{13}$$

This formula, unfortunately, is infinite at each of the vortex centers $\mathbf{x}_i$. This is overcome by the convention that the induced self-velocity of a point vortex, in the absence of boundaries, is zero. The convection equation for vorticity (7) is solved by moving the point vortices at the local fluid velocity:

$$\frac{d\mathbf{x}_i}{dt} = \sum_{j \neq i} \frac{\kappa_j}{2\pi} \frac{(-(y_i - y_j), (x_i - x_j))}{|\mathbf{x}_i - \mathbf{x}_j|^2} \tag{14}$$

Notice that this Lagrangian description of the vorticity field overcomes the difficulty of the nonlinearity in the convection equation.

Thus far we've been considering an unbounded domain. In a region of fluid with stationary boundary $\partial\Omega$, the boundary condition which must be satisfied in this inviscid flow is:

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad on \quad \partial\Omega \tag{15}$$

where $\mathbf{n}$ is the normal to the boundary. (If the boundary is moving at velocity $\mathbf{V}$, the condition becomes $\mathbf{u} \cdot \mathbf{n} = \mathbf{V} \cdot \mathbf{n}$.) This can be satisfied by adding a potential flow to the velocity in (13):

$$\mathbf{u} = \mathbf{u}_\omega + \nabla\phi \tag{16}$$

Taking divergence of this expression, we get:

$$\Delta\phi = 0 \tag{17}$$

$$\partial_n\phi = -\mathbf{u}_\omega \cdot \mathbf{n} \tag{18}$$

A Neumann problem must be solved for ϕ. Depending upon the geometry of the problem, this can be done through the use of conformal mapping where proper image vortices are introduced. [11][19]

The vortex method for incompressible, viscous flow is based on these ideas. The major difference between the Navier-Stokes equations and the Euler equations, besides the diffusion term, is the role of boundaries. The condition at a stationary boundary is now the no-slip condition:

$$\mathbf{u} = 0 \ \ on \ \ \partial\Omega \tag{19}$$

Not only is the normal velocity zero, but also the tangential velocity. Also, unlike inviscid flow, vorticity can be created.

Consider flow past an infinite flat plate with uniform velocity $(U, 0)$ at infinity. The solution of the inviscid equations is $\mathbf{u} = (U, 0)$ everywhere. However, in the viscous case, there must be zero velocity at the plate.

The transition region from the boundary to the uniform flow at infinity (the boundary layer) has thickness on the order of $1/\sqrt{R}$, where R = Reynolds number. Within this boundary layer, vorticity is created. For large Reynolds numbers, the boundary layer is quite small. Here, again, the inadequacy of finite difference schemes for large Reynolds number flow becomes evident. In order to resolve what is going on in the boundary layer, a

minimum number of mesh points is needed within that region. This leads to a prohibitively small mesh width and time step.

Vortex Method for Viscous flow

The Navier-Stokes equations for two-dimensional flow in stream function - vorticity formulation are :

$$\frac{D\omega}{Dt} = \frac{1}{R}\Delta\omega \tag{20}$$

$$\Delta\psi = -\omega \tag{21}$$

$$u = \psi_y \tag{22}$$

$$v = -\psi_x \tag{23}$$

As before, the vorticity field is represented by a discrete collection of vortex elements. In practice, it is necessary to spread the point vortices into "vortex blobs"[11]. We use the approximation suggested by Krasny [13]. The velocity field in an unbounded fluid induced by these modified point vortices would be:

$$\mathbf{u}_\omega = \sum_{i=1}^{m} \frac{\kappa_i}{2\pi} \frac{(-(y-y_i),(x-x_i))}{|\mathbf{x} - \mathbf{x}_i|^2 + \sigma^2} \tag{24}$$

where σ is smoothing parameter.

The no-slip boundary condition on $\partial\Omega$ is split up into two parts:

$$\mathbf{u}\cdot\mathbf{n} = 0 \tag{25}$$

$$\mathbf{u}\cdot\tau = 0 \tag{26}$$

Here n is the normal vector and τ is the tangent vector to $\partial\Omega$. The first condition is satisfied by the introduction of a potential flow, as explained above. The second condition is satisfied by the creation of vortex elements at the boundary with strength $-\mathbf{u}\cdot\tau$ per unit length of

the boundary. (The boundary is partitioned into pieces of length Δs, and a modified point vortex of strength $-2(\mathbf{u}\cdot\tau)\Delta s$ is placed at its center. The factor of two accounts for elements which will immediately fall outside the boundary during a random walk, as explained below.) This numerical creation of vorticity at the boundary mimics what physically occurs.

The vortex algorithm for solution of the Navier-Stokes equations is as follows. Given the centers $\mathbf{x}_i$ and strengths κ_i of the vortex elements at time step n :

1. Satisfy the tangential boundary condition by creating new vortex elements at the boundary.

2. Move all of the vortex elements:

$$\mathbf{x}_i^{n+1} = \mathbf{x}_i^n + \Delta t(\mathbf{u}_\omega^n + \nabla\phi^n) + \eta_i \qquad (27)$$

Here the η_i are independent Gaussian random variables with mean zero and variance $2\Delta t/R$. This random walk accounts for the diffusion term in the vorticity convection equation.

4. Grid-free method with immersed boundary

The vortex method described above is grid-free, and thus the small mesh width restriction of finite difference schemes for large Reynolds numbers is circumvented. We have developed a vortex method to solve the coupled system of a viscous fluid and an immersed elastic boundary. Our treatment of the elastic boundary as a vorticity source is based on McCracken and Peskin [15].

The fluid only feels the presence of the immersed boundary as a singular distribution of external forces:

$$\mathbf{F}(\mathbf{x},t) = \int_B \mathbf{f}(s,t)\delta(\mathbf{x} - \mathbf{X}(s,t))ds \tag{28}$$

Here the integration is over the immersed boundary, and δ is the two-dimensional delta function. We need to examine how this contributes to the evolution of the vorticity. Following McCracken and Peskin [15], we see that the curl of the external force acts as a source of vorticity:

$$\frac{D\omega}{Dt} = \frac{1}{R}\Delta\omega + (\nabla \times \mathbf{F}) \cdot \mathbf{k} \tag{29}$$

Here $\mathbf{k} = (0,0,1)$. We have:

$$(\nabla \times \mathbf{F}) \cdot \mathbf{k} = \int_B [\nabla \times (\delta(\mathbf{x} - \mathbf{X}(s,t))\, \mathbf{f}] \cdot \mathbf{k}\, ds \tag{30}$$

$$= \int_B \nabla\delta(\mathbf{x} - \mathbf{X}(s,t)) \cdot (f_2, -f_1)\, ds \tag{31}$$

where $\mathbf{f} = (f_1, f_2)$. The above integrand can be interpreted as a directional derivative. We discretize this integral as follows:

$$\sum \frac{\delta(\mathbf{x}^+ - \mathbf{X}(s,t)) - \delta(\mathbf{x}^- - \mathbf{X}(s,t))}{2h} |\mathbf{f}| \Delta s \tag{32}$$

where

$$\mathbf{x}^+ = \mathbf{X}(s,t) + h\,\frac{(f_2, -f_1)}{|\mathbf{f}|} \tag{33}$$

$$\mathbf{x}^- = \mathbf{X}(s,t) - h\,\frac{(f_2, -f_1)}{|\mathbf{f}|} \tag{34}$$

Here h is a small parameter.

Therefore, the immersed boundary acts a source of vortex dipoles with dipole moment:

$$(f_2, -f_1)\ \Delta s\ \Delta t \tag{35}$$

We multiply by Δt to get the amount of vorticity to be generated over one time step.

This dipole moment is not, in general, normal to the immersed boundary. McCracken and Peskin [15] split up the force density f into its tangential and normal components:

$$\mathbf{f} = f_\tau \tau + f_\eta \eta \tag{36}$$

where τ is the unit tangent to the immersed boundary curve, and η is the unit normal. It can be shown, using integration by parts, that the normal component contributes a monopole layer of vorticity along the boundary with strength density:

$$\frac{d}{ds}\left[\frac{f_\eta(s)}{|\mathbf{X}'(s)|}\right] \tag{37}$$

The tangential component contributes a dipole layer as described above. The process of splitting up the creation of vortex elements into a monopole layer and dipole layer oriented normal to the boundary adds more numerical stability to the calculations.

We now outline our basic vortex method in the plane within which there is an immersed boundary. The state of the system at time $t = n\Delta t$ is given by the configuration $\mathbf{X}_k^n$ of material points of the immersed boundary, and the discrete collection of modified point vortices centered at x_j^n with strengths κ_j^n. The basic algorithm is:

1. Compute force density $\mathbf{f}_k^n$ using the current boundary configuration.

2. Use (1) to create new vortex elements at the boundary.

3. Move each of the vortex elements in the velocity field induced by the other vortex elements plus a random walk to simulate diffusion.

4. Compute the velocity at each of the points of the immersed boundary using (24). Move each of these at this local fluid velocity.

This algorithm has the advantage of being grid-free, and since it has no *real* boundaries, we needn't worry about the introduction of image vortices to satisfy the normal boundary condition discussed in the previous section.

Explicit evaluation of the force density at the current boundary configuration produces large instabilities. The reason for this is that the boundary is stiff and the forces are large enough to make the boundary overshoot equilibrium in one time step. We use the approximate-implicit method described by Peskin [16]. In general when boundary points are coupled to each other through the forces, this approximate-implicit method requires the solution of a nonlinear optimization problem. However, in the examples addressed in this report, our immersed boundary points act as *independent* springs. Therefore, step (1) is performed very quickly.

In step (2), we either create two or three vortex elements per boundary point. This depends on whether we wish the dipole layer to be normal to the boundary - in which case we create both a dipole and a monopole layer of vorticity.

Steps (3) and (4) require the computation of velocities of the vortex elements and boundary points respectively using the current positions of vortex elements and their strengths. We use a second order Runge Kutta method to update the positions of the vortex elements and the boundary points.

5. Computational Results

Flow past a cylinder

Our first test problem will be flow past a circular cylinder. The cylinder is modelled by a collection of immersed boundary points which are tethered to fixed spatial positions by springs of resting length zero. We are deliberatly choosing a test problem where the equilibrium position of the immersed boundary is known and is not time dependent. Flow past a cylinder has been well-studied since the qualitative features of the flow (such as vortex shedding) are Reynolds number dependent. For a detailed discussion of the simulation of flow past a cylinder using vortex methods, the reader is referred to Cheer [4]. Our simulations differ in that the cylinder is represented as a singular external force field in the fluid, and not a computational boundary.

We found that introducing the dipole layer normal to the cylinder made the calculations more stable. Therefore, at each time step, we create three vortex elements per boundary point. This is shown in Figure 1 for an immersed cylinder made up of forty immersed boundary points. The cylinder has a diameter $d = .1\ cm$ and it placed in a uniform flow $U = 2\ cm/sec$. We used a time step $\Delta t = .00025\ sec$, and spacing between boundary points of $\Delta s = .1\pi/40\ cm$. The numerical parameter which represents the distance of the dipole vortex elements from the immersed boundary is $h = .8\Delta s$. The viscosity was chosen to be 10^{-8}. The smoothing parameter $\sigma = .75\Delta s$. The Reynolds number based upon the uniform flow velocity and diameter of the cylinder is therefore on the order of 10^7. The tethering stiffness constant used was 10^6.

Figures 2 and 3 show velocity fields of the flow about the cylinder at time $t = .0125 sec$. We show two space scales to emphasize the fact that because this is a grid-free method, the velocities may be calculated from the vortex element configuration at any spatial points by using (24). We are therefore able to resolve the boundary layer with a much finer degree of accuracy than if we were working on a fixed grid. Smoothing does take place, however, both

in the representation of the vortex blob functions (parameter σ) and in the finiteness of the dipole layer (parameter h).

Stagnation points at the upstream and downstream centerline of the cylinder can be seen in Figures 2 and 3. The magnitude of the velocity vectors at the top and bottom of the cylinder are larger than the free stream velocity, as is to be expected. The flowfield within the cylinder is nearly zero, which is evidence that the tethering forces are doing their job. However, these flowfields do not yet show vortex formation and shedding at the cylinder. We have not run this code long enough to exhibit these features, which become evident at such a high Reynolds number. In order for the approximate-implicit force calculations to remain stable, we need to use a relatively small time step. This causes the creation of an extremely large number of vortex elements in a small real-time simulation. We will discuss some remedies to this in Section 6.

Flow past a flat plate

In this example, we will simulate incompressible, inviscid flow about a flat plate which is oriented normal to the free flow. There are two possible inviscid flows which are solutions. One is a continuous potential flow which is steady and symmetric with respect to the plate. However, the velocities near the ends of the plate are infinite. Another solution is an unsteady, discontinuous potential flow, which sheds vorticity from the plate. Here, the velocities at the ends of the plate are finite. This solution, when compared with experiments, is more realistic and can be thought of as the vanishing viscosity limit. [14]

The flat plate is represented by an elastic boundary which is made up of forty immersed boundary points. The plate lies on the line $x = 1$ and goes from $y = 1$ to $y = 3$. It is tethered to space with springs of resting length zero and stiffness constants 10^7. A uniform background flow of $U = .5$ is imposed. The numerical parameters used are $\Delta s = .05$, $h = .4\Delta s$, and $\sigma = .75\Delta s$.

To modify the algorithm in order to model invisid flow, we discard the random walk which approximates diffusion. Moreover, we only force the *normal* component of velocity of

the plate to coincide with that of the fluid. (Our immersed boundary point can therefore only move in the $x-$ direction.) In inviscid flow, tangential slip is allowed.

In these simulations, we are only going to create a single dipole layer of vorticity at the plate ,i.e. two vortex elements per boundary point are created at each time step. This dipole layer is actually tangent to the plate.

Figure 4 shows velocity fields about the plate at $t = .04, t = .25, t = .5$, and $t = .75$. The first flow field appears to be symmetric about the plate; a wake has not yet had a chance to form. The successive flow fields show the formation of two counter-rotating vortices at the ends of the plate. These are getting bigger as time goes on.

Our numerical scheme has picked out the unsteady, inviscid solution. We believe this happens because our velocity field, due to the smoothing parameter σ, is not allowed to be infinite anywhere. Moreover, the finiteness of our dipole layer h adds some smearing which is manifested as numerical viscosity.

The dipole layer of vortex elements is initialized to lie tangentially along the plate, except for the two elements which are placed a distance of h from the top and the bottom. The vortex elements on the plate move approximately with the same velocities as the tethered boundary points, and thus, approximately remain on the plate. However, vortex elements are shed from the top and bottom. Figure 5 shows the location of vortex elements at time $t = .04$ and at $t = .25$. Figure 6 shows a contour plot of vorticity at $t = .25$. These figures qualitatively compare to the vortex sheet roll-up described in Krasny [14].

6. Conclusions and future work

Finite difference schemes do not perform well in simulations of high Reynolds number flow because a restrictive number of grid points must be used to resolve boundary layers. We have presented a grid-free numerical method which may be used to calculate flows around an immersed elastic boundary. We have presented numerical results in two simple cases, where the boundary motion is specified and is not time dependent. Our initial results show that the method gives the correct qualitative features of the desired flow.

The major drawback of this algorithm is the growth of the number of vortex elements at each time step. The computation of the vortex-vortex interactions becomes very expensive. Presently, we are performing the straightforward $O(m_n^2)$ operations, where m_n is the total number of vortex elements present at time step n. There are, however, techniques which can be used to speed up this calculation: Anderson's method of local corrections [1], or the multipole method of Greengard and Rokhlin [12]. We are planing to incorporate one of these in our code.

As the algorithm stands, m_n is always increasing by two or three times the number of boundary points each time step. In an exterior problem, like flow past a flat plate, we can throw away vortex elements when they move a certain distance from the immersed boundary. However, in interior calculations, like blood flow in the heart, some vortex elements are trapped inside the immersed boundary. We need to investigate the systematic merging of nearby vortex elements in order to make large time simulations feasible.

We need to establish the dependence of this numerical method on the numerical parameters. These are:

1) The equilibrium distance between boundary points Δs.

2) The small parameter h which determines the distance of the dipole layer from the boundary.

3) The parameter σ used in smoothing the velocity field near the center of the vortex blobs.

We would like to show convergence of this numerical scheme as these parameters are refined. We should also include another parameter, κ_{max} which would be the maximum strength any one vortex element can carry. If a strength $\kappa_i > \kappa_{max}$ is calculated, that vortex element ought to be split up into other elements of smaller strengths, which add up to κ_i. This has shown to be very important in the convergence of standard vortex methods [21].

The numerical examples presented here were chosen for their simplicity. These immersed boundaries do not undergo time dependent motions, and their equilibrium position in space has been specified by specifying tether points. Moreover, each boundary point is uncoupled from its neighbors in the force density calculations. Once a better understanding of the numerical method is achieved on these test problems, we can use it, with very minor modifications, to model time dependent problems in biofluiddynamics.

References

[1] C.R. ANDERSON, J. Comput. Phys., 62 (1986), p. 111.

[2] J.B. BELL, P. COLELLA and H.M. GLAZ, J. Comput. Phys., 85 (1989), p. 257.

[3] R.P. BEYER, submitted for publication (1990).

[4] A.Y. CHEER, SIAM J. Sci. Stat. Comp., 4, (1983), p. 685.

[5] A.J. CHORIN, Math. Comp., 22 (1968), p. 745.

[6] A.J. CHORIN, J. Fluid Mech., 57 (1973), p. 785.

[7] L.J. FAUCI, J. Comput. Phys., 86 (1990), p. 294.

[8] L.J. FAUCI, submitted for publication (1991).

[9] L.J. FAUCI and C.S. PESKIN, J. Comput. Phys., 77 (1988), p. 85.

[10] A.L. FOGELSON, J. Comput. Phys., 56 (1984), p. 111.

[11] A.F. GHONIEM and Y. CAGNON, J. Comp. Phys., 74, (1988), p. 283.

[12] L. GREENGARD and V. ROKHLIN, J. Comp. Phys., 73, (1987), p. 325.

[13] R. KRASNY, J. Comp. Phys., 65, (1986), p. 292.

[14] R. KRASNY, ASME Fed - International Symposium on Nonsteady Fluid Dynamics, 92, (1990), p. 449.

[15] M.F. MCCRACKEN and C.S. PESKIN, J. Comp. Phys., 35, (1980), p. 183.

[16] C.S. PESKIN , J. Comp. Phys., 25, (1977), p. 220.

[17] C.S. PESKIN and D.M. MCQUEEN, J. Comp. Phys., 81, (1989), p. 372.

[18] C.S. PESKIN and D.M. MCQUEEN, J. Comp. Phys., 82, (1989), p. 289.

[19] C.S. PESKIN and A.W. WOLFE, Federation Proc., 37, (1978), p. 2765.

[20] S. SAMN, submitted for publication (1991).

[21] J.A. SETHIAN and A.F. GHONIEM, J. Comp. Phys., 74, (1988), p. 283.

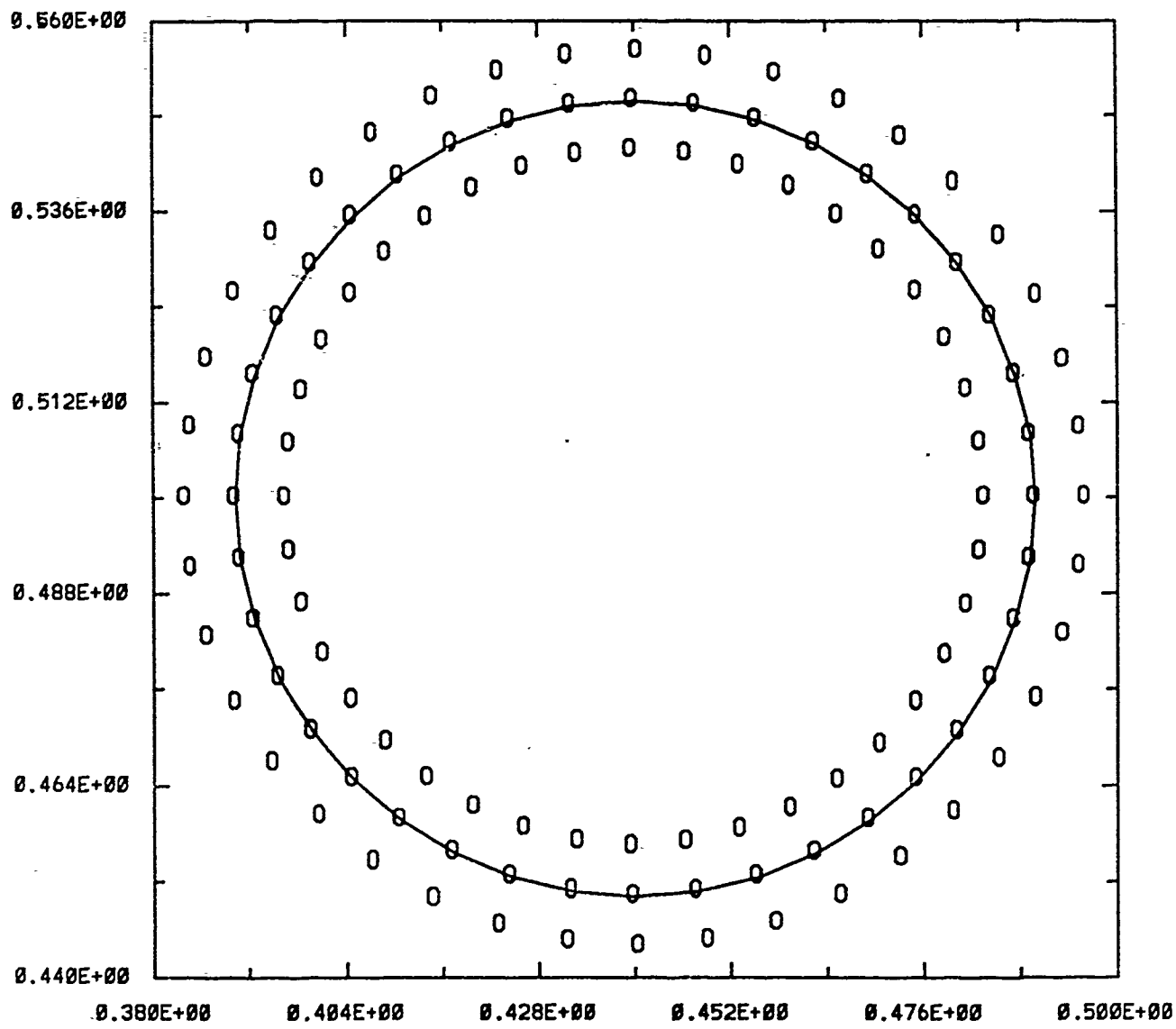

POSITION OF VORTEX ELEMENTS
0.560E+00
0.536E+00
0.512E+00
0.488E+00
0.464E+00
0.440E+00
0.380E+00
0.404E+00
0.428E+00
0.452E+00
0.476E+00
0.500E+00

FIGURE 1

FLOW PAST CYLINDER

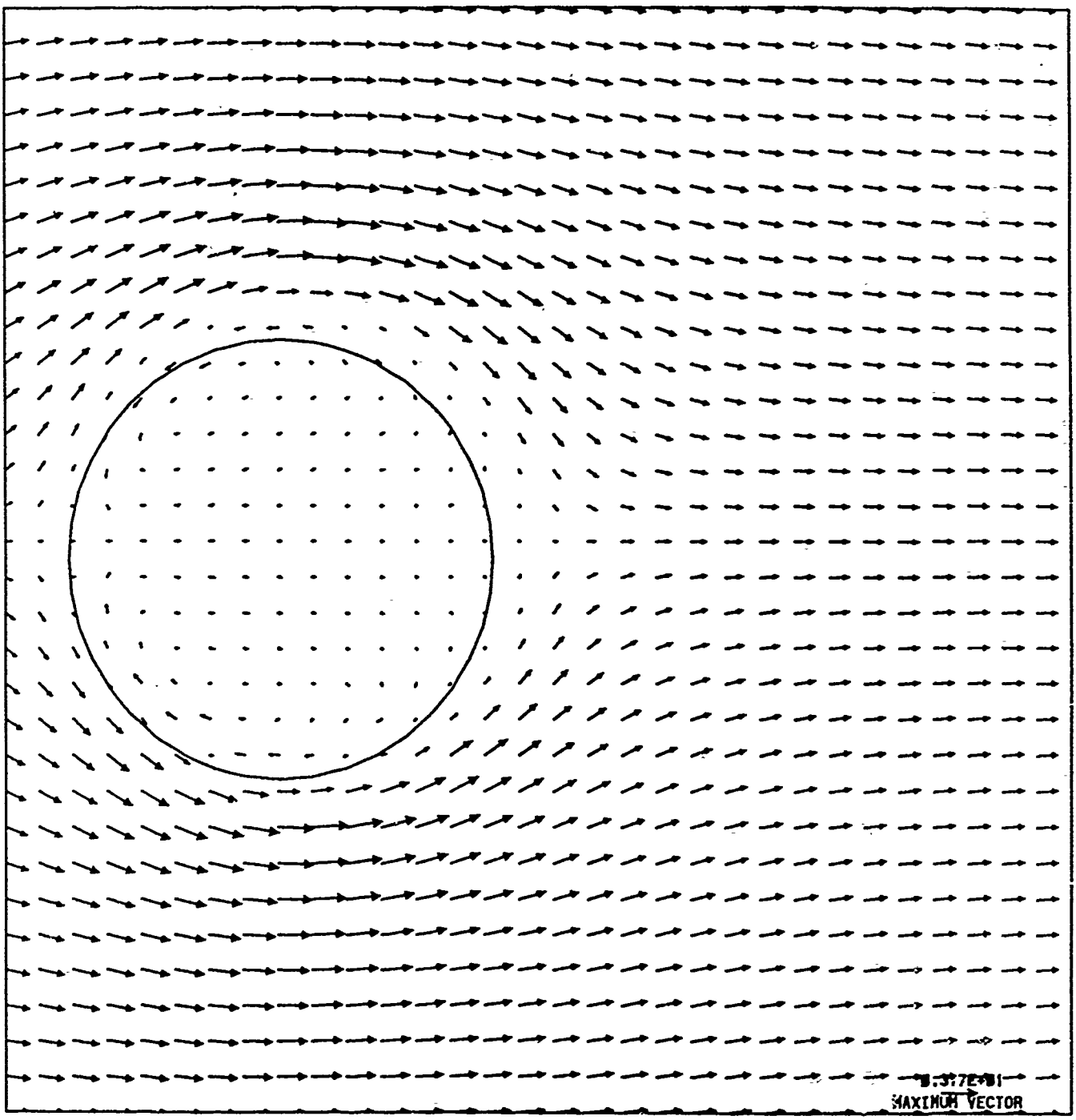


FIGURE 2

FLOW PAST CYLINDER

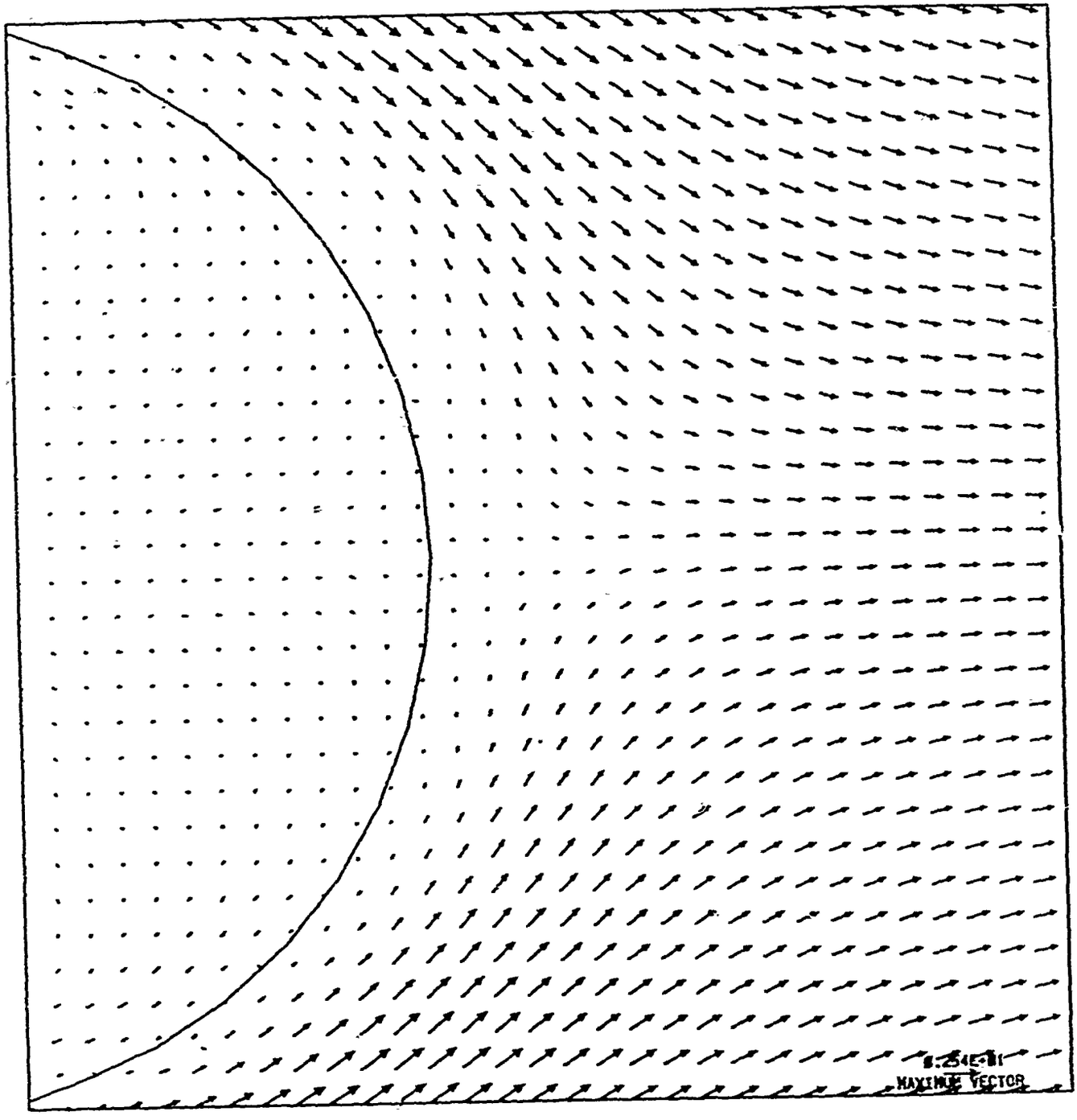


FIGURE 3

FLOW PAST FLAT PLATE

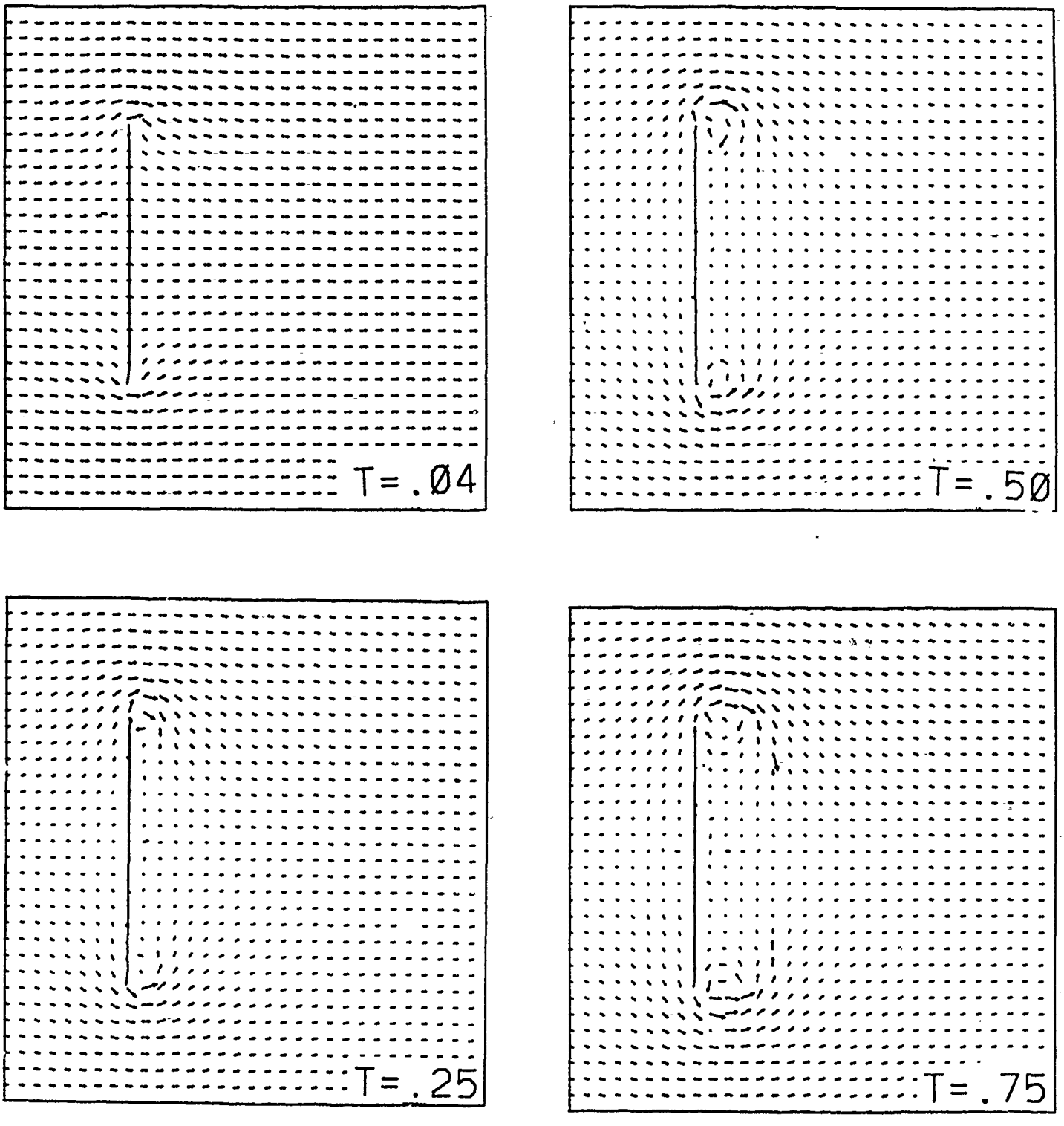
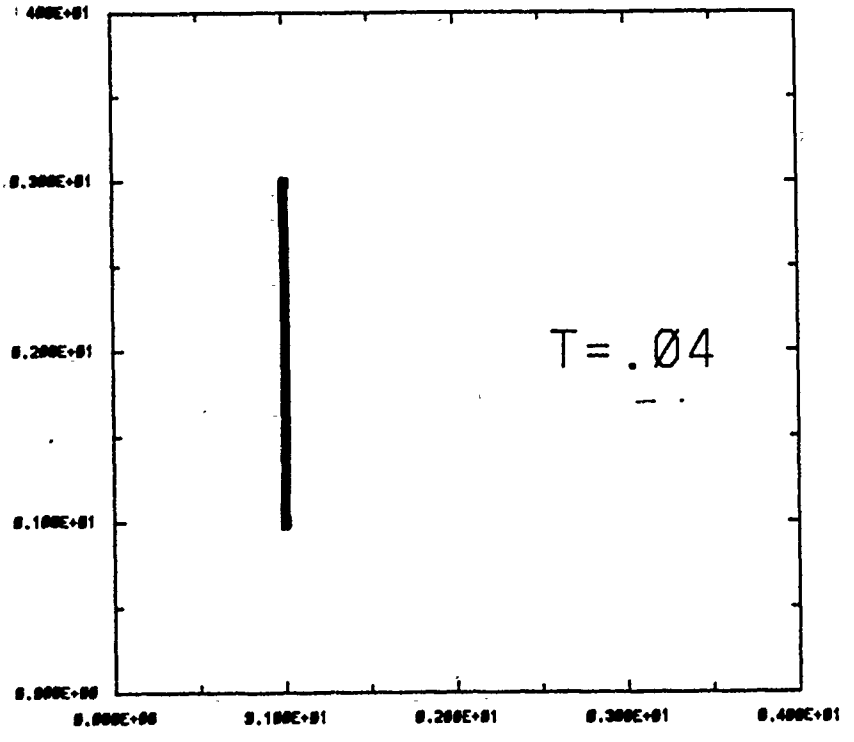


FIGURE 4

POSITION OF VORTEX ELEMENTS



POSITION OF VORTEX ELEMENTS

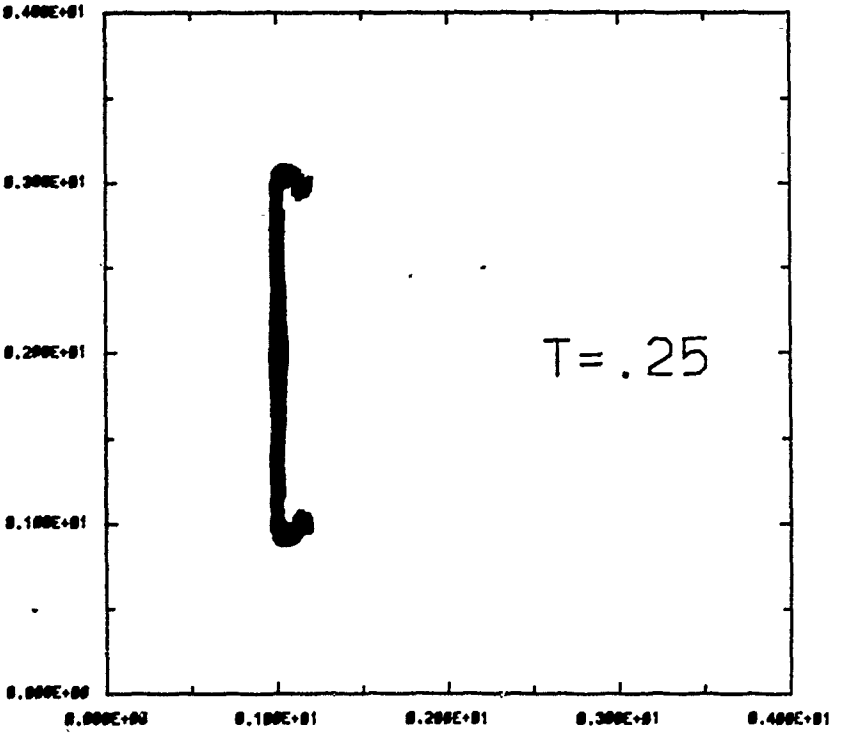


FIGURE 5

CONTOURS OF VORTICITY - T=.25

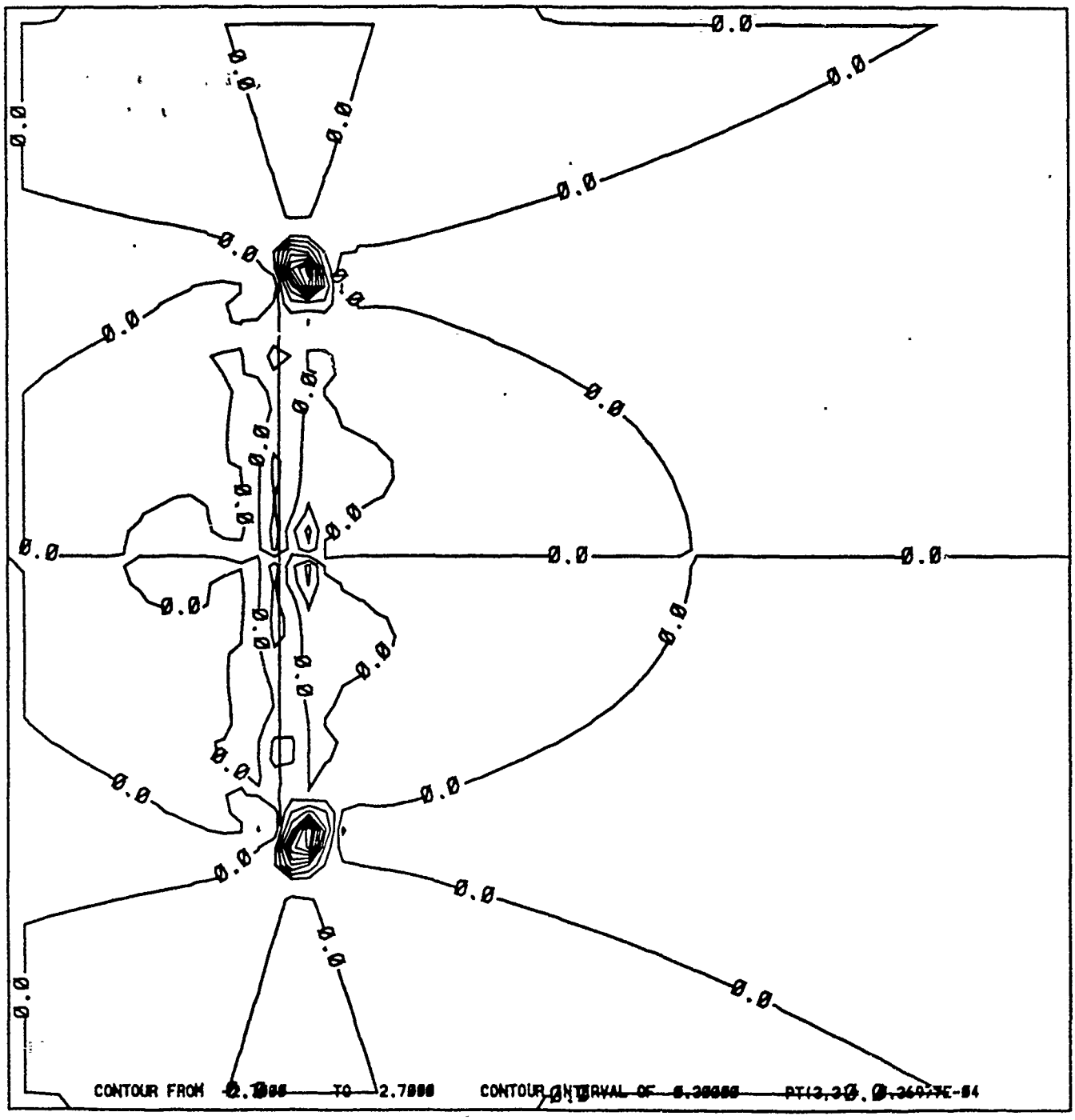


FIGURE 6